

Network Coding Tree Algorithm for Multiple Access System

Zhengchuan Chen*, Ke Xiong[†], Pingyi Fan* and Chen Chen*

*Tsinghua National Laboratory for Information Science and Technology (TNList),
and Department of Electronic Engineering, Tsinghua University, Beijing, China.

[†]School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China.

*{chenzc10@mails., fpy@c-c10@mails.}@tsinghua.edu.cn, [†]kxiong@bjtu.edu.cn

Abstract—Network coding is famous for significantly improving the throughput of networks. The successful decoding of the network coded data relies on some side information of the original data. In that framework, independent data flows are usually first decoded and then network coded by relay nodes. If appropriate signal design is adopted, physical layer network coding is a natural way in wireless networks. In this work, a network coding tree algorithm which enhances the efficiency of the multiple access system (MAS) is presented. For MAS, existing works tried to avoid the collisions while collisions happen frequently under heavy load. By introducing network coding to MAS, our proposed algorithm achieves a better performance of throughput and delay. When multiple users transmit signal in a time slot, the mixed signals are saved and used to jointly decode the collided frames after some component frames of the network coded frame are received. Splitting tree structure is extended to the new algorithm for collision solving. The throughput of the system and average delay of frames are presented in a recursive way. Besides, extensive simulations show that network coding tree algorithm enhances the system throughput and decreases the average frame delay compared with other algorithms. Hence, it improves the system performance.

Index Terms—Network coding, Tree algorithm, Multiple access system.

I. INTRODUCTION

The medium access control (MAC) protocols play an important part in modern communication networks such as Internet and wireless local area network (WLAN). Different protocols share the multiple access medium among users in different way. They are usually divided into two types. The first type is called a reservation protocol, in which users make reservations in the first stage in order to determine the transmission scheduling in the next stage. The second type is called a direct transmission protocol, in which users send their frames following certain rules based on the channel feedback. The feedback often indicates that zero, one or two and more frames (collision state) are transmitted in the last slot [1]. The core of both types of protocol rests with the collision solving algorithms which dictate how to solve the collision situation.

Network coding has brought underlying change to the communication world [2] [3]. It was first proposed to achieve the capacity of the multicast network. Later, the extended version of this discovery—the physical layer network coding

(PHY NC) was applied to many practical situations such as two-way relay networks [4], vehicular communications [5] and butterfly networks [6]. PHY NC shows great potential in improving the performance of these networks [7].

The typical examples of collision solving algorithm include the slotted ALOHA algorithm and binary tree algorithm. It is well known that the slotted ALOHA algorithm has a limit throughput of $\frac{1}{e}$ and also requires some kinds of stabilizing method [8] [9]. In order to avoid these defects, the binary tree algorithm is proposed later [10]. It is proved to be able to maintain stability and increase the achievable throughput [11]. Several modifications are made to the binary tree algorithm to achieve a larger throughput. In both binary tree algorithm and its extended version, the access point (AP) of the multiple access system (MAS) drops the sum signal of the collided frames. It is often assumed that combination of the collision frames is useless because the frames interfere with each other. The thought brought by PHY NC, which makes use of the collision frames, can be used to improve the performance of classical collision solving algorithms. After successful decoding some raw component of the collided frames, one of the original frame of the collided frames can be decoded successfully by regarding the received raw component frames as side information. It is shown that both the throughput and delay¹ of slotted ALOHA algorithm are improved combining with PHY NC [14] due to that multiple received copies introduce a diversity gain. However, in [14] the authors assumed that the AP can know which terminals collide. This assumption is too strong to implement the slotted ALOHA algorithm into MAS. In [15], the authors considered to combine the network coding strategy with ALOHA algorithm while simplifies the physical channel model as Galois Field addition and complication. Because the ALOHA algorithm dominates the main performance essentially, the throughput of the algorithm proposed in [15] always holds a increase-decrease characteristic in terms of the arrival rate. As binary tree algorithm is more efficient than ALOHA algorithm when system is under heavy load [16], we combine PHY NC with the binary tree algorithm to achieve even greater performance in this paper.

The rest of the paper is organized as follows. In Section II, we introduce the system model and claim some necessary

¹An extension of this work with theoretical analysis on the throughput and delay of the network coding tree algorithm has been submitted.

¹It is noted that delay is another important performance metric in wireless communications, [12] and [13] and reference therein include examples of delay-considered optimal protocol design for other wireless scenarios

preliminaries. In Section III, we propose a network coding tree algorithm (NCTA). In Section IV, we compare the throughput and average delay of NCTA to those of binary tree algorithm (BTA) and ALOHA system by simulation. Conclusions are drawn in Section V.

II. SYSTEM MODEL AND PRELIMINARIES

A. system model

Let us consider a MAS with finite m users and only one AP. Assume the system runs with slots and the synchronization among the users and the AP is guaranteed. In each slot, one user can transmit a frame if other users keep silent. If two or more users transmit their frame in the same slot, signals are collided and the AP cannot decode any frame. At the end of each slot, we assume that users are able to get a feedback via listening the channel or from the AP. For instance, in a bus system, AP is nothing different from a user from a physical layer viewpoint. The assumption that users can obtain the channel feedback via listening is reasonable. In a WLAN system, limited bandwidth can provide necessary channel feedback from AP to users. In this paper, we further consider that each user can know from the feedback that a collision, or a successful transmission, or nothing happens in the last slot. For clarity, let us use a uniform function to represent the feedback and neglect the physical layer system setup. Denote the signal received at AP by x . We define

$$f(x) = \begin{cases} 0, & x \text{ made up of noise} \\ 1, & x \text{ includes one frame} \\ e, & \text{otherwise} \end{cases} \quad (1)$$

where e is used to represent the state when collision is occurring.

The frame arrivals are assumed to be independent Poisson flows. Let us denote the sum arrival rate of the system by λ . Accordingly, the arrival rate at each user is λ/m while frames are poisson arrived. Usually, in MAS, if the frame collides with others, it will be retransmitted in later slots until it is received successfully. We will extend this rule and reduce the retransmission times by deploying signal cancellation which realizes network decoding. In most of the existing works, users were assumed to have an infinite long buffer. To focus on planting network coding into MAS and investigating the throughput-delay performance gain induced by network coding idea, we try to simplify the system model in this paper by considering that the buffer length is one. That is, the buffer is modeled as a binary state machine². If there is a frame at the user, it is in 'active' state, otherwise it is in 'inactive' state.

To compare the performance with other algorithms, throughput and average delay are regarded as the two main performance metrics of interest in this work. On one hand, throughput stands for the synthesized system efficiency and it is the performance of interest in most cases. On the other hand, under the throughput metric, the average delay play

an important role in describing the quality of service (QoS) provided by a system [17]. For clarity, we use Φ and D to represent them in our analysis, respectively.

To analyze the MAS embedded with PHY NC, we may first review some classical collision solving algorithms.

B. Time Division Multiplexing

In time division multiplexing (TDM), time slots are assigned to users. Each user has to wait for his own time slot to transmit frames. There is no frame collision under this scheme. System is expected to transmit a frame in each time slot under heavy load. Hence, TDM achieves very well performance with a large arrival rate. The highest throughput can even approach 1. Nevertheless, for light load, many time slots are idle and wasted since there is no user to transmit. Moreover, the fixed transmission order of TDM also induces a large average delay for all arrival rate $\lambda \in [0, 1)$. The average delay, D_{TDM} , of TDM is about to $\frac{m}{2}$ [16].

C. Slotted ALOHA Protocol

In order to diminish the delay of the MAS, ALOHA scheme was presented which allows users to compete for accessing the channel. If frames collide in a time slot, each of them will be retransmitted after a random delay with a probability p . In the light load case, less collisions occur so that the large delay endured in TDM scheme is significantly alleviated. Nevertheless, even in the moderate load case, the frequent collision may push the delay up and the system then goes to an instable status along with increasing data arrival rate dramatically. In low arrival rate case, ALOHA has better performance. Moreover, as it can be performed in a distributed manner, it is widely employed in lots of network scenarios.

D. Binary Tree Algorithm

Instead of letting all the users with data competing for the channel in each time slot, BTA only allows part of the users to retransmit their data frames in the following slots to diminish the collision probability[16]. Once a collision occurs, the users are randomly split into subsets. For each subset, users retransmit their frames in a future slot. Dividing is along with collision. If no collision happens for a time slot, the frame transmitted by some user in the subset will be successfully decoded or none of the users has frames to transmit[16]. BTA bridges ALOHA and TDM in terms of the throughput and average delay performance. In BTA, mixed signals of collided frames are dropped by AP. To utilize the information contained in collided signals, we propose the NCTA in the following.

III. NETWORK CODING TREE ALGORITHM

NCTA utilizes the mixed signal of the collided frames as a physical layer network coded signal. Again, dividing follows collision, either in a random or a fixed manner. The successful frame receiving of a subset helps it decode frames of sibling subsets via canceling signal of the successfully decoded frames from the collided mixed signals. As only the status is fed back by AP, including free time slot, collision and successful

²This assumption is used to simplify the theoretical analysis of the throughput and average delay, which is strongly related to the probability of empty buffer at each node. We will investigate the general case with finite buffer length L in another paper.

transmission, after canceling the decoded signal from the mixed signals, it is feasible for AP to accurately feed back the status of the rest of the signals by observing the mixed signals. So, the AP may know no frame, one frame, or two or more data frames included in the rest of the mixed signals.

To analyze Φ and D , we take two types of system waiting time into consideration. In the first type, the buffer observes only one time slot, that is, the buffer only keeps the frames arrived in the last slot. We call this system setup as *System Type I*. In the other case, waiting time is defined as the number of time slots used for solving the last collision corresponding to the root node of the algorithm tree. This system setup is called as *System Type II*. Let us denote the system waiting time by W .

To describe the algorithm clearly, we present the pseudo code as Algorithm 1.

Note that $y' - y_1$ can be regarded as the signal transmitted by nodes in A_2 , polluted by noisy of transmission corresponding to A' and that of transmission corresponding to A .

IV. SIMULATIONS

We present the throughput and average delay of the new proposed network coding tree algorithm and other known algorithms with respect to the arrival rate λ for two types of system in this section. To compare NCTA, BTA with ALOHA, we modify the ALOHA system to guarantee that the system conditions for these algorithms are the same. We state the modified ALOHA system as follows.

At the beginning, the system is with unblocked state. If there are frames arriving at some users, then those users transmit frame. If no collision occurs, the system maintains unblocked state, otherwise it goes into blocked state. In blocked state, all the users with frames transmit in each time slot with probability p until the frame is successfully received by AP. After that, system returns to unblocked state. In the modified ALOHA algorithm, if there are n users with frame, then p is set to $\frac{1}{n+1}$. Because it can maximize the probability of successful transmission, namely, $C_n^1 p(1-p)^n$ in next time slot. In the rest of this paper, we use ALOHA to represent the modified ALOHA system without ambiguity.

First, for *System Type I*, we compare the throughput of NCTA, BTA and ALOHA. We consider $M = 8$ and the arrival rate of the system, $\lambda \in [0, 2]$. For every point, it is the average of 5000 simulations. As illustrated in Fig. 1, for all the arrival rate $\lambda \in [0, 2]$, the throughput of NCTA is clearly greater than that of BTA and ALOHA. Among them, the throughput of NCTA ascends in terms of λ and keeps stable around 0.75 while that of NC and ALOHA, keeps 0.5. That is, the throughput is increased by 0.25. Average delays of different algorithms corresponding to this setup are depicted in Fig. 2. Again, for all the arrival rate, the average delay of NCTA is less than that of BTA and ALOHA while that of BTA and ALOHA are roughly the same with each other.

In summary, under *System Type I*, both throughput and average delay of NCTA are better than those of BTA and ALOHA.

For *System Type II*, to keep consistent with that of *System Type I*, we assume $M = 8$ and $\lambda \in [0, 2]$ again. For every

Algorithm 1 Network Coding Tree Algorithm

Initialization:

Set waiting time $W = 1$; Clear all the user buffers; Initialize two new stacks S_1 and S_2 ; Denote all the users by A

Iteration:

```

1: while  $A \neq \emptyset$  do
2:   All the users with non-empty buffer transmit; Denote
   the sum signal by  $y$ ;
3:   if  $f(y) \neq e$  then
4:     Set  $W = 1$ ;
5:     Break;
6:   else
7:      $S_1 \leftarrow A$ ;
8:      $S_2 \leftarrow y$ ;
9:   end if
10:  while  $S_1$  is not empty do
11:     $A' \leftarrow$  Pop an element from  $S_1$ ;
12:     $y' \leftarrow$  Pop an element from  $S_2$ ;
13:    Uniformly divide  $A'$  into two subsets  $A_1$  and  $A_2$ ;
14:    All the users in  $A_1$  transmit frame in next time slot;

15:  Denote the sum signals received at AP by  $y_1$ .
16:  if  $f(y_1) \neq e$  AND  $f(y' - y_1) = e$  then
17:     $S_1 \leftarrow A_2$ ;
18:     $S_2 \leftarrow y' - y_1$ ;
19:  else
20:    if  $f(y_1) = e$  AND  $f(y' - y_1) \neq e$  then
21:       $S_1 \leftarrow A_1$ ;
22:       $S_2 \leftarrow y_1$ ;
23:    else
24:      if  $f(y_1) = e$  AND  $f(y' - y_1) = e$  then
25:         $S_1 \leftarrow A_2$ ;
26:         $S_1 \leftarrow A_1$ ;
27:         $S_2 \leftarrow y' - y_1$ ;
28:         $S_2 \leftarrow y_1$ ;
29:      end if
30:    end if
31:  end if
32: end while
33: Set  $W$  according to the time slots used for collision
   solving.
34: end while

```

point, we average out 5000 simulations. The throughput and average delay of the three algorithms are shown in Fig. 3 and Fig. 4. In Fig. 3, we can see the maximal throughput of the modified ALOHA system is 0.45 and keeps 0.4 when throughput is stable, which is greater than that of the slotted ALOHA, $\frac{1}{e} \approx 0.37$. The throughput of BTA, is about 0.5 while that of NCTA, is significantly approaching 1 when λ is increasing. Before it achieves the maximum, the throughput of the three algorithms are increasing linearly with the arrival rate. After that, the throughput of ALOHA drops slightly. This shows that NCTA and BTA are more stable than ALOHA.

Comparing Fig. 1 with Fig. 3, it is not hard to see that for *System Type II*, the throughput of ALOHA decreases while

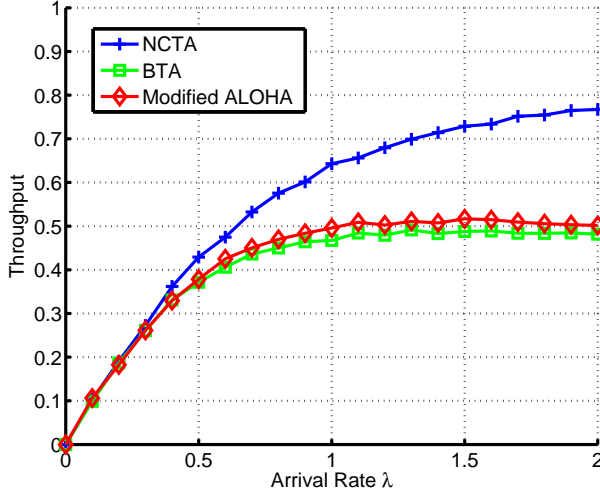


Fig. 1. The throughput of network coding tree algorithm (NCTA), binary tree algorithm (BTA) and modified ALOHA system for system type I.

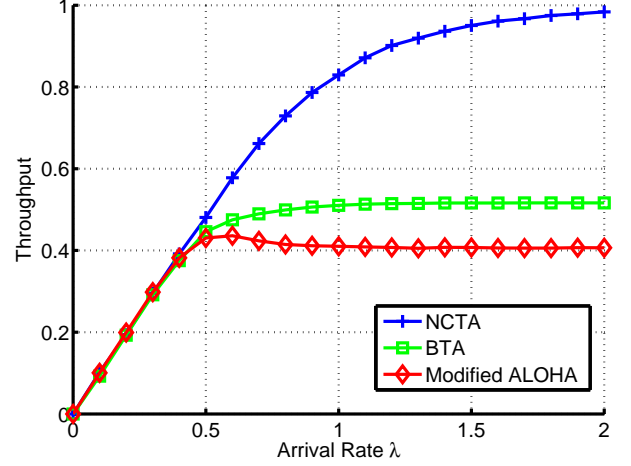


Fig. 3. The throughput of network coding tree algorithm (NCTA), binary tree algorithm (BTA) and modified ALOHA system for system type II.

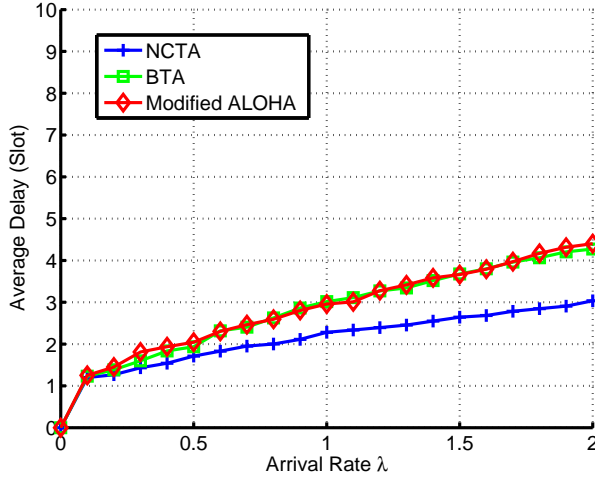


Fig. 2. The average delay of network coding tree algorithm (NCTA), binary tree algorithm (BTA) and modified ALOHA system for system type I.

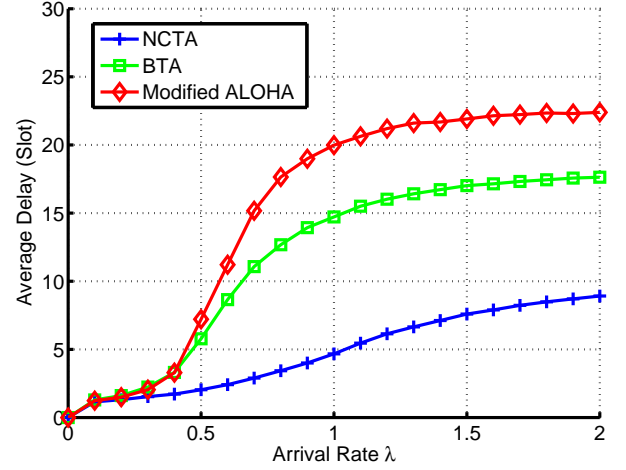


Fig. 4. The average delay of network coding tree algorithm (NCTA), binary tree algorithm (BTA) and modified ALOHA system for system type II.

that of BTA improves a little and that of NCTA increases significantly. Under *System Type II*, the system waiting time is longer, hence, the probability of collision becomes larger. This implies that ALOHA system is inefficient to solve collision while NCTA is good at solving collisions with many frames.

Fig. 4 shows the average delay of three algorithms under *System Type II*. Along with the increasing of λ , all the average delay curves keep ascending. For all the arrival rate, the average delay of NCTA is always less than those of BTA and ALOHA. Besides, when $\lambda = 0.4$, the slope of the delay curves increase dramatically. The slope roughly reflects the stability of the algorithm. From Fig. 4, we can see that the slope of ALOHA's average delay is the largest one followed by that of BTA and that of NCTA is the smallest. Comparing with the delay curves of *System Type I*, systems bear a larger delay for all the three algorithms. This is induced by the increased quantity of collided frames, hence, the increased time slots

used for collision solving.

Next, we consider the impact of system user quantity under system type II. In Fig. 5 and Fig. 6, the throughput and the average delay of NCTA for $m = 8$, $m = 16$, $m = 32$ are illustrated. In Fig. 5, we can see that for given arrival rate, the throughput of the system is increased slightly along with the increasing user number. It is predicted that when $m \rightarrow +\infty$, the throughput will stable around a fixed value. Moreover, the income induced by increasing the user number is marginal. In Fig. 6, it is not difficult to see that along with the increasing user number m , the average delay of the frames is blowing up. The slope of the delay curve is ascending dramatically. The reason is that when system has more users, the collided frames are more than that of a system with less users. there are many frames involved in a collision, therefore, the time taken to solve the collision will be longer. In average, each frame has to wait long time to be decoded successfully by AP. According to Fig. 5 and Fig. 6, when user number falls down, the throughput decreases very slightly while the average

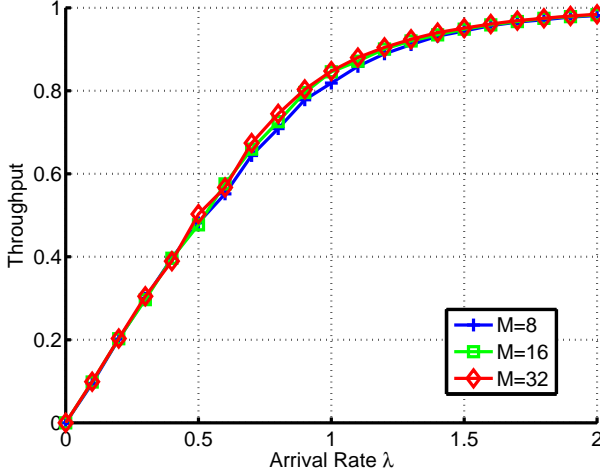


Fig. 5. The throughput of network coding tree algorithm for system type II. M represents the system user number.

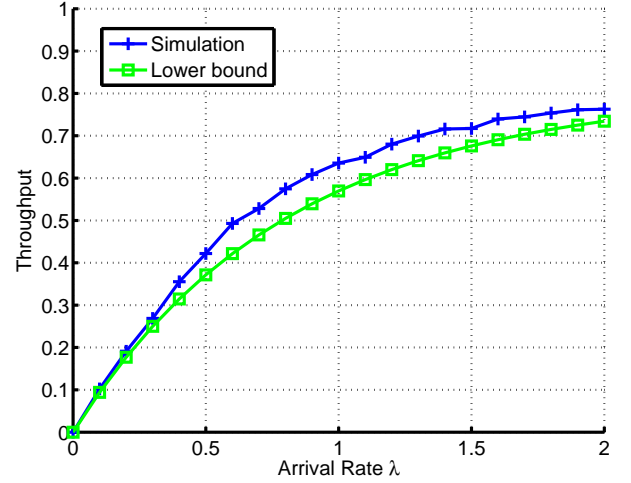


Fig. 7. The simulation result and lower bound for the throughput of network coding tree algorithm, system type II.

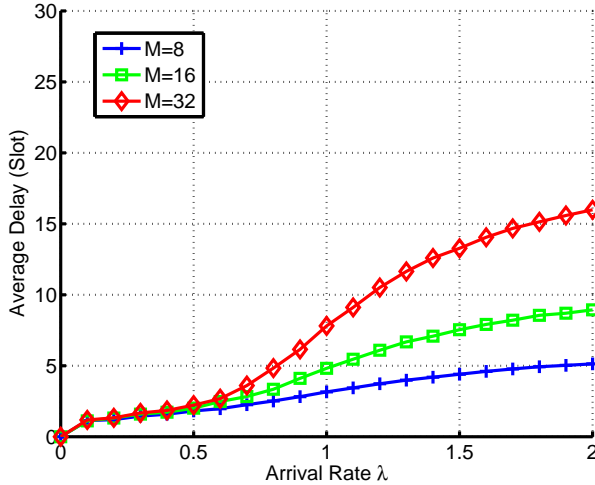


Fig. 6. The average delay of network coding tree algorithm for system type II. M represents the system user number.

delay of the frames, drops down drastically.

V. CONCLUSION

In this work, we introduced network coding into multiple access system and propose a network coding tree algorithm. The implicit form of the throughput and average delay of the system operated with the new algorithm were given in an iterative way. To show the performance of the algorithm, we compare the throughput and average delay induced by the new algorithm with that of modified ALOHA scheme and tree algorithm via simulation. The results showed that when operated with network coding tree algorithm, multiple access system can achieve a larger throughput while the frames bear shorter average delay. Besides, when system has more users, the throughput increases slightly while the average delay ascends drastically.

REFERENCES

- [1] B. S. Tsybakov, V. A. Mikhailov, "Free Synchronous Packet Access in a Broadcast Channel with Feedback," *Problemy Peredachi Inform.*, vol. 14, No. 4, pp. 32-59.
- [2] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1204-1216, July 2000.
- [3] T. Ho, D. Lun, "Network Coding: An Introduction," *Cambridge University Press*, U.K. 2008.
- [4] S. Zhang, S. C. Liew, and P. Lam, "Physical-Layer Network Coding," *MobiCom 2006*, September, 2006, California.
- [5] Q. Wang, P. Fan, K. B. Letaief, "On the joint V2I and V2V scheduling for cooperative VANETs with network coding," *IEEE Transactions on Vehicular Technology*, vol. 61 no. 1, pp. 62-73, 2012.
- [6] Z. Chen, P. Fan, "Multiple Multicast for Half-duplex Butterfly Network: A Deterministic Approach", *Wiley Wireless Communications and Mobile Computing*, Wiley Online Library, 2014.
- [7] S. C. Liew, S. Zhang, L. Lu, "Physical-layer network coding: Tutorial, survey and beyond," *Physical Communication*, vol. 6, pp. 4-42, 2013.
- [8] N. Abramson, "The Aloha System! Another Alternative for Computer Communications," *Proc. 1970 Fall Joint Computer Conference (AFIPS Conference)*, 1970:37
- [9] N. Abramson, "Development of the Alohanet," *IEEE Transactions on Information Theory*, vol. 31, No. 3, pp. 119 - 123, Mar. 1985.
- [10] J. Capetanakis, "Generalized TDMA: The Multi-Accessing Tree Protocol," *IEEE Transactions on Communications*, vol. 27, No. 10, pp. 1476-1484, 1979.
- [11] J. Capetanakis, "Tree Algorithms for Packet Broadcast Channels," *IEEE Trans. Inform. Theory*, vol. 25, No. 5, pp. 505-515, May 1979.
- [12] H. Zhou, P. Fan, J. Li, "Global proportional fair scheduling for networks with multiple base stations," *IEEE Transactions on Vehicular Technology*, vol. 60 no. 4, pp. 1867-1879, 2011.
- [13] Q. Wang, D. O. Wu, P. Fan, "Delay-constrained optimal link scheduling in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 59 no. 9, pp. 4564-4577, 2010.
- [14] Cocco G., Ibars C., Gunduz D. et al, "Collision Resolution in Slotted ALOHA with Multi-User Physical-Layer Network Coding," *Proc. IEEE 73rd Vehicular Technology Conference*, pp. 1-4, 2011.
- [15] Ni Rui, W. Zhou, G. Wei, "NC-ALOHA: A Novel MAC Protocol," *Proc. 11th IEEE Singapore International Conference on Communications Systems*, pp. 1135-1139, 2008.
- [16] R. G. Gallager, "Data Network," *Cambridge University Press*, U.K. 2008.
- [17] L. Kleinrock, F. Tobagi, "Packet Switching in Radio Channels: Part I-Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics," *IEEE Transactions on Communications*, vol. 23, No. 12, pp. 1400-1416. Dec. 1975.
- [18] T. M. Cover, J. A. Thomas, "Elements of Information Theory," *John Wiley & Sons*, 2ed, 2012.